

## **IN THE CLAIMS**

### **Claims pending**

- At time of the Action: Claims 1-40
- After this Response: Claims 1-40

**Canceled or Withdrawn claims:** Claims 14, 24, 30, 39 (all currently) and  
41-74 (previously)

**Amended claims:** Claims 1-2, 4-5, 7-13, 15, 18-23, 25-29, 31-38 and 40

**New claims:** None

1. **(Currently Amended)** Interfaces, stored on one or more computer-readable media, to be called on kernel transaction management objects, comprising:

application program interfaces (APIs) to implement operations on a kernel transaction object (TX), ~~at least one~~the TX representing a transaction, the TX ~~and~~ being accessible ~~to~~by at least one process participating in the transaction;

APIs to implement operations on a kernel resource management object (RMO), ~~at least one~~the RMO representing a relationship between a TX associated with ~~a corresponding~~the transaction manager and at least one resource that participates in the transaction, the resource being an entity capable of storing data in a durable state; and

APIs to implement operations on a kernel enlistment (EN) object, ~~at least one~~the EN representing a relationship between a resource manager and the transaction.

2. **(Currently Amended)** Interfaces according to Claim 1, wherein each of the APIs to implement operations on the TX, the RMO, and the EN utilize a handle to refer to an object.

3. **(Original)** Interfaces according to Claim 2, wherein each of the handles is an opaque reference to a unique object.

4. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to transmit pre-prepare messages to resource managers associated with a transaction.

5. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to transmit a prepare request to resource managers enlisted in a transaction.

6. **(Original)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for a new TX to be created for a transaction.

7. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for ~~an existing~~the TX to be opened for a transaction.

8. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to commit a transaction.

9. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to abort a transaction.

10. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to save a current state of the transaction.

11. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to retrieve information about the TX for a requestor.

12. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to set information.

13. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the TX to close.

14. **(Canceled)**

15. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for a new RMO to be created.

16. **(Original)** Interfaces according to Claim 15, wherein the new RMO is volatile.

17. **(Original)** Interfaces according to Claim 15, wherein the new RMO is durable.

18. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for ~~an existing~~the RMO to open for a transaction.

19. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the RMO to be destroyed.

20. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the RMO to transmit information regarding the RMO to a requestor.

21. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the RMO to set information.

22. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for the RMO to be enlisted on a transaction at least once.

23. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for a notification from a resource manager for the RMO.

24. **(Canceled)**

25. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-is to implement operations on the TX by the RMO.

26. **(Currently Amended)** Interfaces according to Claim 25, wherein the at least one of the APIs ~~API~~-is to inform the TX that pre-preparing is complete.

27. **(Currently Amended)** Interfaces according to Claim 25, wherein the at least one of the APIs ~~API~~-is to inform the TX that transaction preparation has been completed by a requested resource manager.

28. **(Currently Amended)** Interfaces according to Claim 25, wherein the at least one of the APIs ~~API~~ is to inform the TX that a resource manager has completed rolling back a transaction.

29. **(Currently Amended)** Interfaces according to Claim 25, wherein the at least one of the APIs ~~API~~ is to inform the TX that a resource manager has committed to a transaction.

30. **(Canceled)**

31. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~ calls for a resource manager to be registered as a communications resource manager for a particular protocol.

32. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~ calls for a representation of a transaction to be serialized into a buffer.

33. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~ calls for information representing registered protocols to be serialized into a buffer.

34. **(Currently Amended)** Interfaces according to Claim 32, wherein at least one of the APIs ~~API~~-calls for a transaction represented by the serialization be made available by a transaction management object.

35. **(Currently Amended)** Interfaces according to Claim 2, wherein at least one of the APIs ~~API~~-calls for a transaction to be propagated to a destination using push-style propagation.

36. **(Currently Amended)** Interfaces according to Claim 35, wherein at least one of the APIs ~~API~~-calls for the output of the API calls for the transaction to be propagated to a destination using push-style propagation to be retrieved.

37. **(Currently Amended)** Interfaces according to Claim 31, wherein at least one of the APIs ~~API~~-is called when transaction propagation has been completed.

38. **(Currently Amended)** Interfaces according to Claim 31, wherein at least one of the APIs ~~API~~-is called when a requested transaction propagation has failed.

39. **(Canceled)**

40. **(Currently Amended)** An apparatus for implementing a transaction, comprising:

a kernel transaction object (TX) to represent a transaction, the TX  
~~and being accessible to~~ by at least one process participating in the  
transaction;

a kernel resource manager object (RMO) to represent a relationship  
between a TX associated with a ~~corresponding~~ the transaction manager and  
at least one resource that participates in the transaction, the resource being  
an entity capable of storing data in a durable state; and

a kernel enlistment object (EN) to represent a relationship between a  
resource manager and the transaction,

wherein two-phase commit processing is executed by calling APIs  
on the TX, the RMO, and the EN.

41-74. **(Previously Canceled)**